Maximizing OpenGL Performance

Session 214

These are confidential sessions—please refrain from filming or taking pictures.

Graphics and Media

Maximizing OpenGL Performance

John Stauffer
OpenGL Engineering Manager

http://developer.apple.com/wwdc2005/

OpenGL Optimization Overview

- Graphics seed
- Key concepts
- Basic tips
- Texture uploads
- Vertex throughput

Mac OS X OpenGL Seeding Program

- Tighter feedback loop with you!
- Access prerelease versions of OpenGL and drivers
  - Early access to features
  - Active testing and feedback
  - Performance analysis
  - Bug regression
- Send request to openglseed@apple.com
  - Note: space is limited

Key Concepts

- OpenGL is a state machine
  - The hardware has a ONE current state
- The OpenGL state is compiled in programs
  - Programs are cached
- Switching to a new state is expensive

Key Concepts

- Use tools to analyze performance
  - OpenGL profiler
  - OpenGL driver monitor
  - Shark
- OpenGL extensions
  - Optimizing data throughput
  - Optimize memory footprint

Basic Tips

- Call `glFlush` only for select cases
- Never call `glFinish`
- Use OpenGL extensions to optimize data flow
  - `GL_ARB_vertex_buffer_object`
  - `GL_APPLE_texture_range`, `GL_APPLE_client_storage`
- Minimize OpenGL state changes

Demo

Vertex performance

John Stauffer
OpenGL Engineering Manager

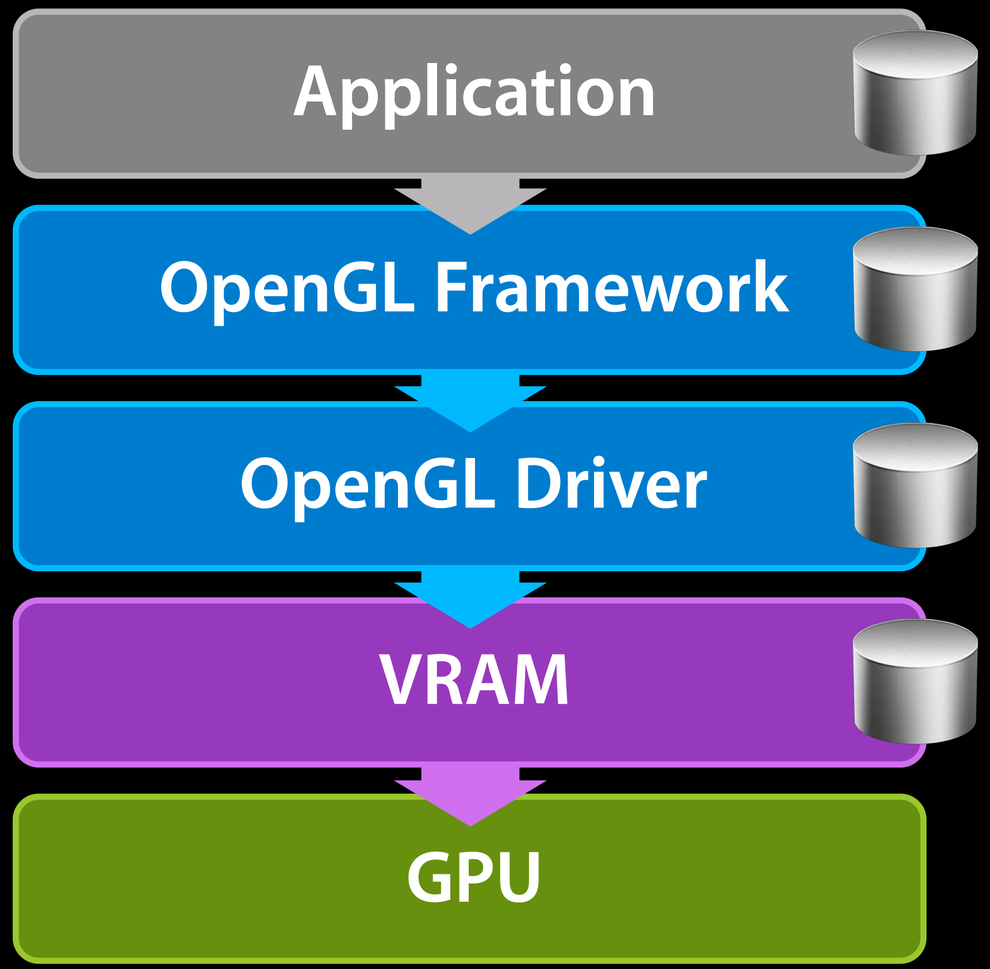
http://developer.apple.com/wwdc2005/

Texture Uploads

John Stauffer
OpenGL Engineering Manager

http://developer.apple.com/wwdc2005/

Texture Upload Overview



Sample Code: Texture Range

Texture Performance Demo

John Stauffer
OpenGL Engineering Manager

http://developer.apple.com/wwdc2005/

Texture Optimization Basics

- Minimize CPU copies and conversions
  - Correct setup will minimize CPU copies
- Optimal texture types and formats

```
GL_BGRA, GL_UNSIGNED_INT_8_8_8_8_REV
GL_BGRA, GL_UNSIGNED_SHORT_1_5_5_5_REV
YCBCR_422_APPLE, UNSIGNED_SHORT_8_8_REV_APPLE
```

Sample Code: Texture Range

OpenGL Texture Extensions

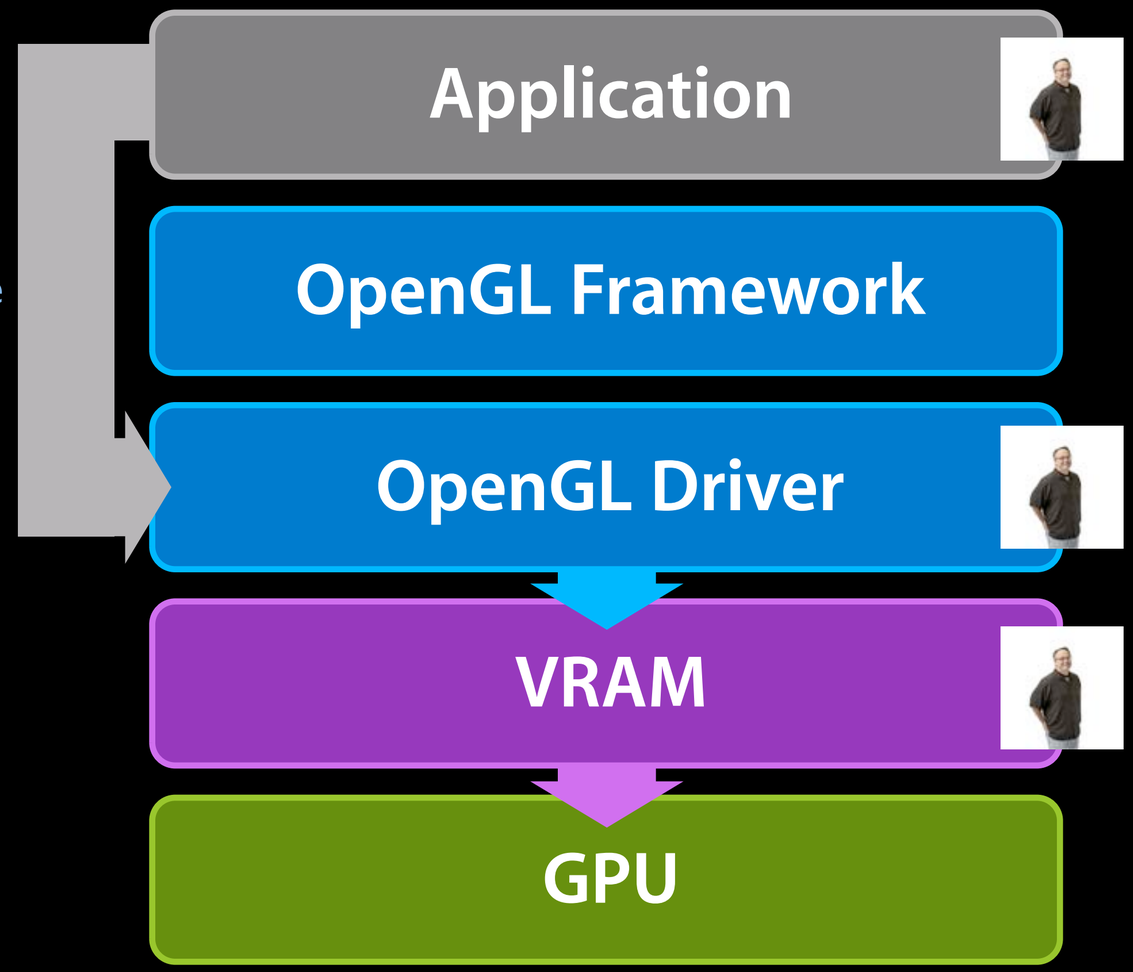
- `GL_APPLE_client_storage`
  - Requires application to retain texture storage
- `GL_APPLE_texture_range`

  `GL_STORAGE_CACHED_APPLE`

  `GL_STORAGE_SHARED_APPLE`
- `GL_EXT_texture_rectangle`
  - Required for some hardware for direct DMA

Sample Code: Texture Range

Client Storage



GL_APPLE_client_storage

Sample Code: Texture Range

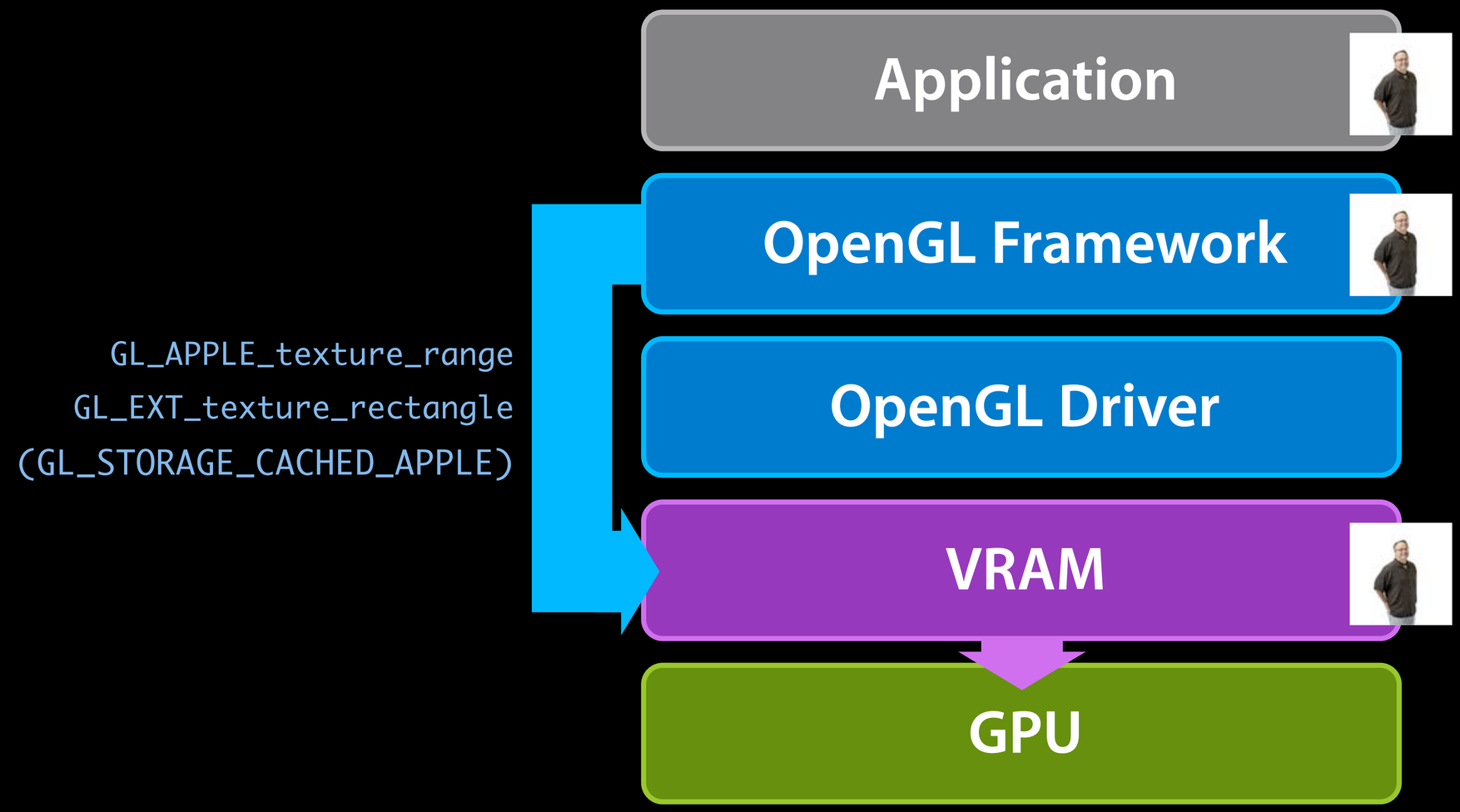
Client Storage

Usable for power-of-two and non power-of-two

```
glPixelStorei(GL_UNPACK_CLIENT_STORAGE_APPLE, GL_TRUE);
```

Sample Code: Texture Range

Texture Range and Rectangle Texture



Sample Code: Texture Range

Texture Range and Rectangle Texture

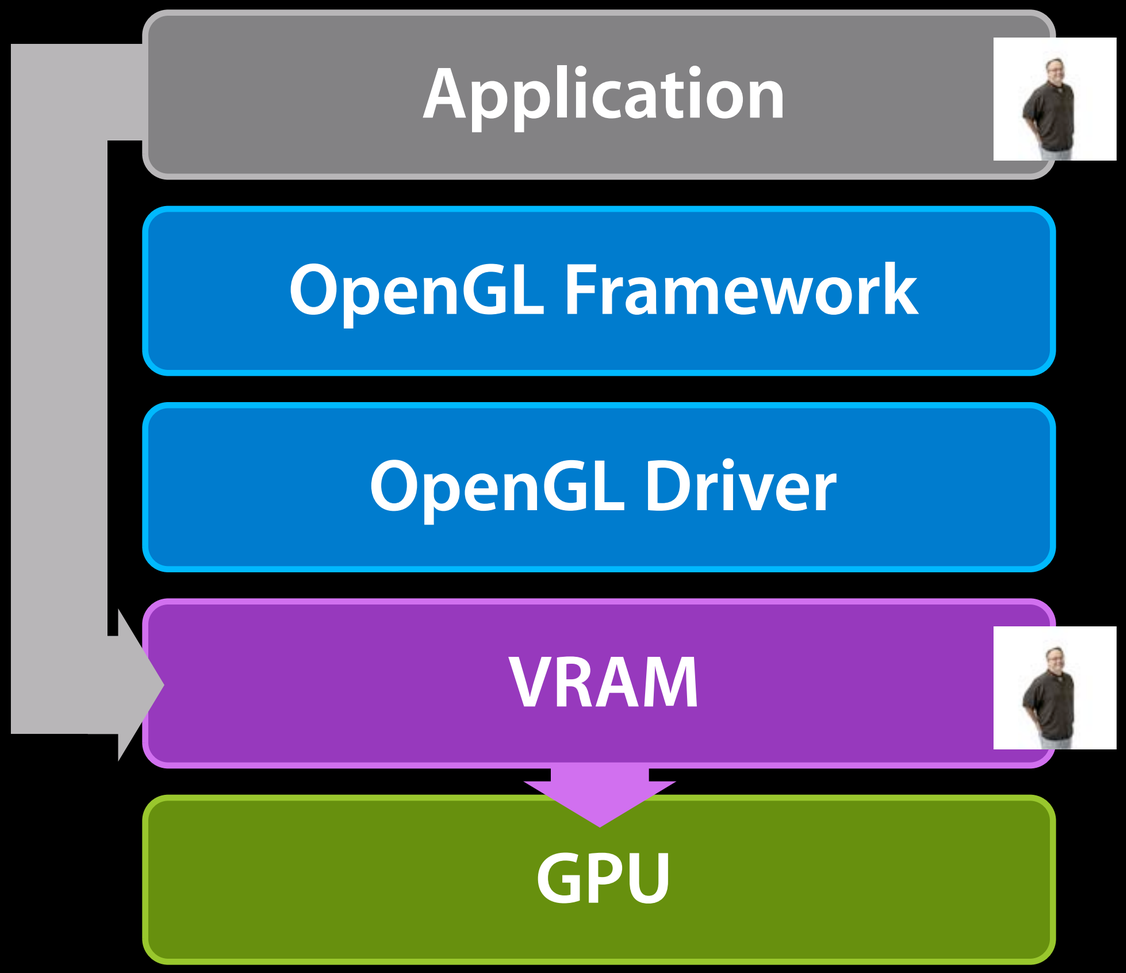
Texture range cached hint

```
glTexParameteri
(GL_TEXTURE_RECTANGLE_EXT,
  GL_TEXTURE_STORAGE_HINT_APPLE,
  GL_STORAGE_CACHED_APPLE);
```

Sample Code: Texture Range

OpenGL Texture Extensions

GL_APPLE_client_storage
GL_APPLE_texture_range
GL_EXT_texture_rectangle
(GL_STORAGE_CACHED_APPLE)



Sample Code: Texture Range

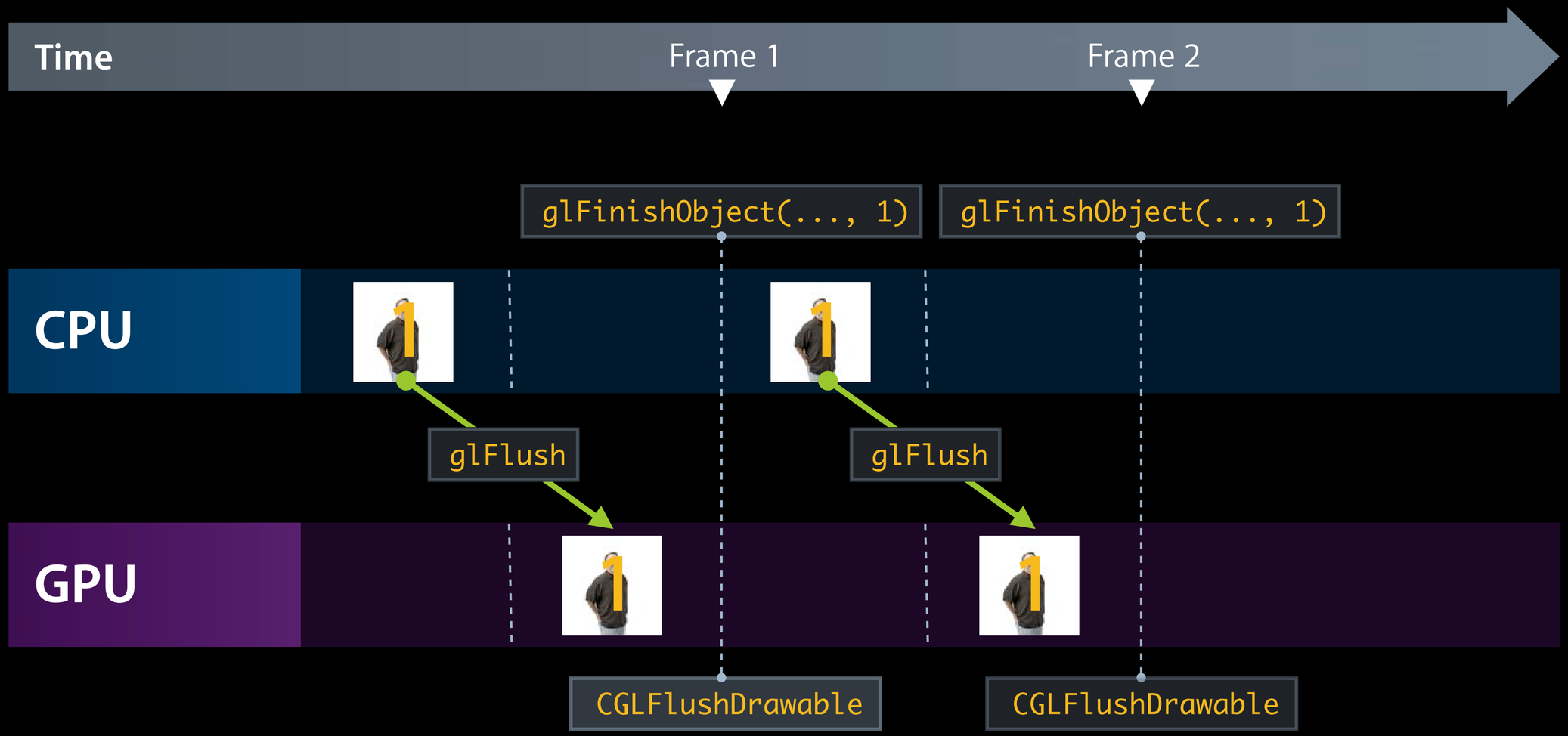
OpenGL Texture Extensions

Texture setup for non power-of-two (cached)

```
glBindTexture(GL_TEXTURE_RECTANGLE_EXT, id);

glTexParameteri(GL_TEXTURE_RECTANGLE_EXT,
  GL_TEXTURE_STORAGE_HINT_APPLE,
  GL_STORAGE_CACHED_APPLE);

glPixelStorei(GL_UNPACK_CLIENT_STORAGE_APPLE, GL_TRUE);

glTexImage2D(GL_TEXTURE_RECTANGLE_EXT,
  0, GL_RGBA, sizex, sizey, GL_BGRA,
  GL_UNSIGNED_INT_8_8_8_8_REV,
  image_ptr);
```

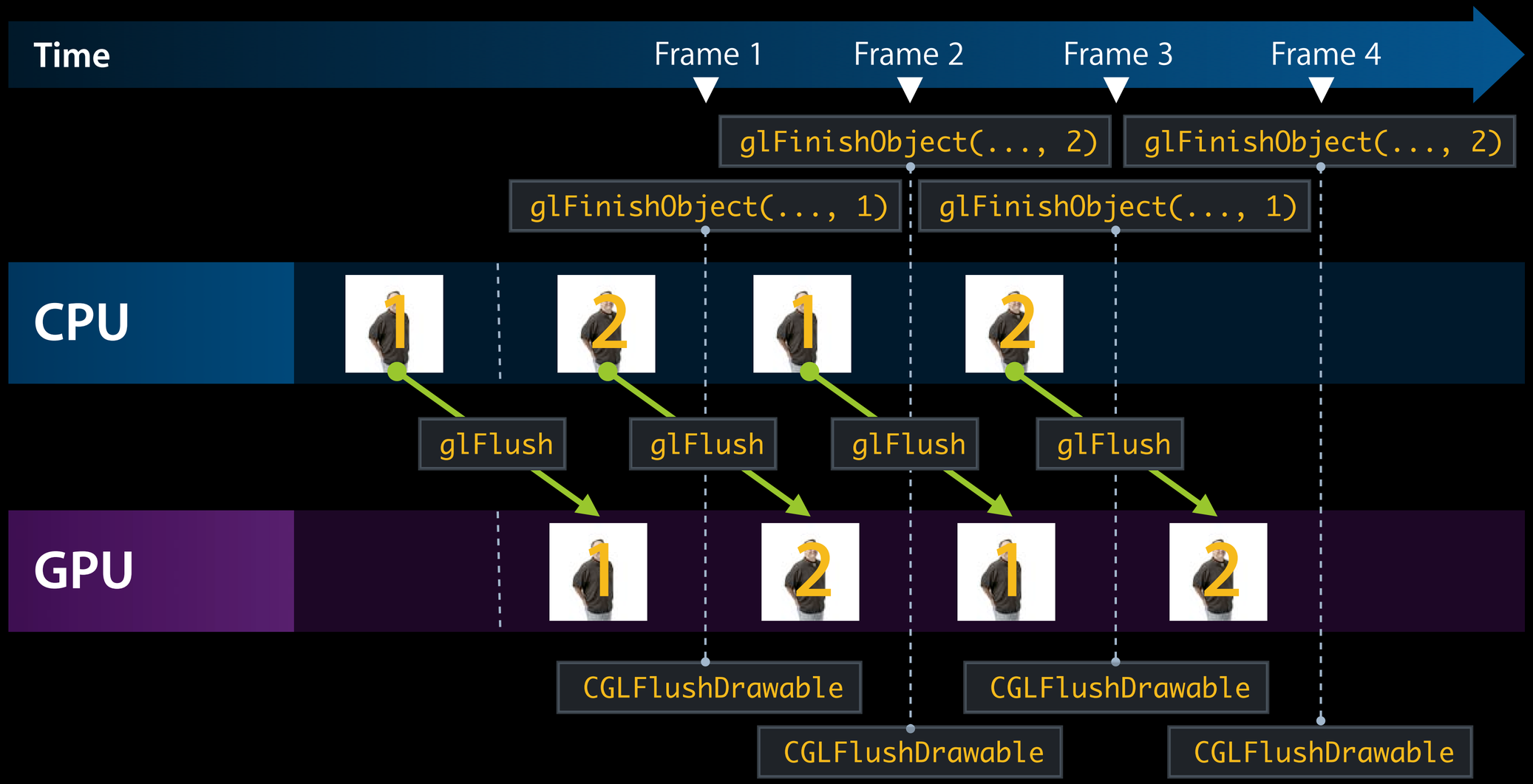
Sample Code: Texture Range

Non Double-Buffered Texture Data



Sample Code: Texture Range

Double-Buffered Texture Data



Sample Code: Texture Range

Texture Range Synchronization

- OpenGL commands
  - `GL_APPLE_fence`
    - Test for token completion
    - Test for object completion

      `GL_TEXTURE`

Sample Code: Texture Range

GL_APPLE_fence

Testing a fence object for completion

```
glFinishObjectAPPLE(GL_TEXTURE, name);
```

Sample Code: Texture Range

Texture Performance Demo

John Stauffer
OpenGL Engineering Manager

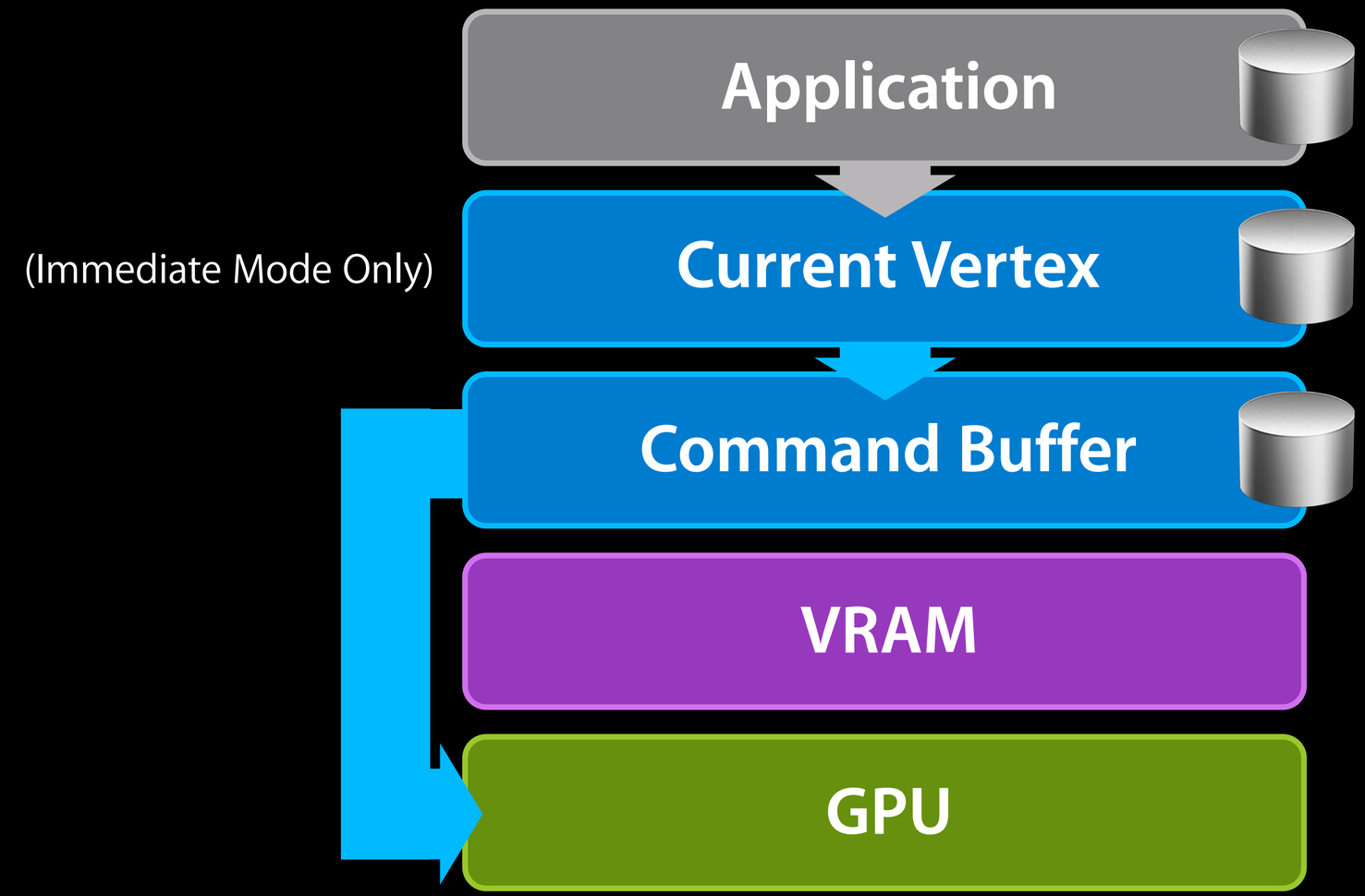
http://developer.apple.com/wwdc2005/

Vertex Throughput

John Stauffer
OpenGL Engineering Manager

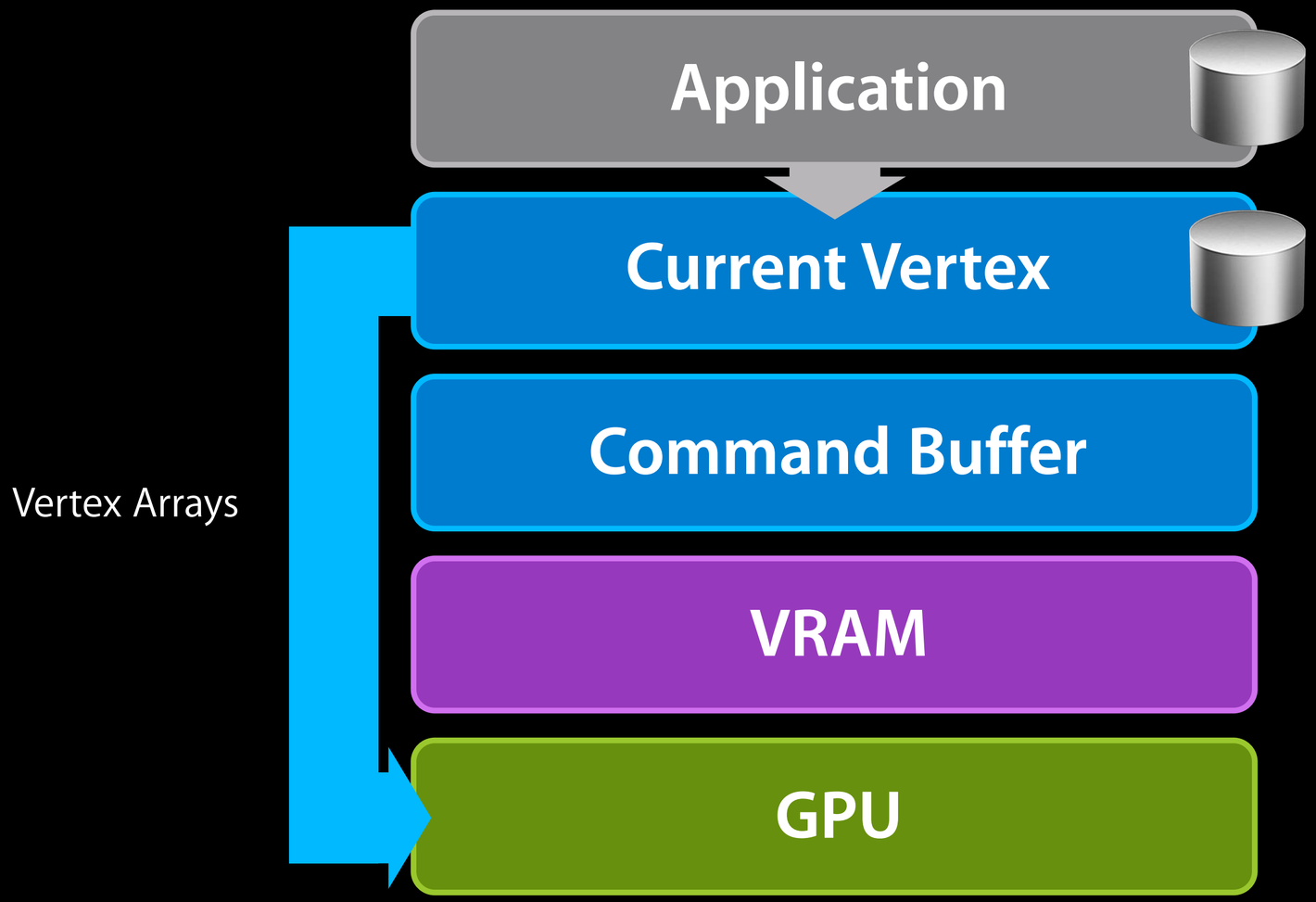
http://developer.apple.com/wwdc2005/

Vertex Pipeline Overview



Sample Code: Texture Range

Vertex Arrays



Vertex Arrays

Sample Code: Texture Range

Vertex Optimization Basics

- Minimize CPU type conversions
  - Use `GLfloat`, `GLshort`, or `GLubyte` types for vertex data
- Minimize function call overhead
  - Use vertex arrays

    `glDrawRangeElements`
    `glDrawElements`
    `glDrawArrays`
  - Use CGLMacros.h
- Maximize vertices per draw command
- Use vertex programs to offload CPU work

Sample Code: Vertex Performance Test

Vertex Optimization Basics

- Use extensions to eliminate CPU copies

  `GL_ARB_vertex_buffer_object`
- Cache static data in VRAM
- Use display lists for static data
  - Caches data in VRAM
  - Define complete vertices

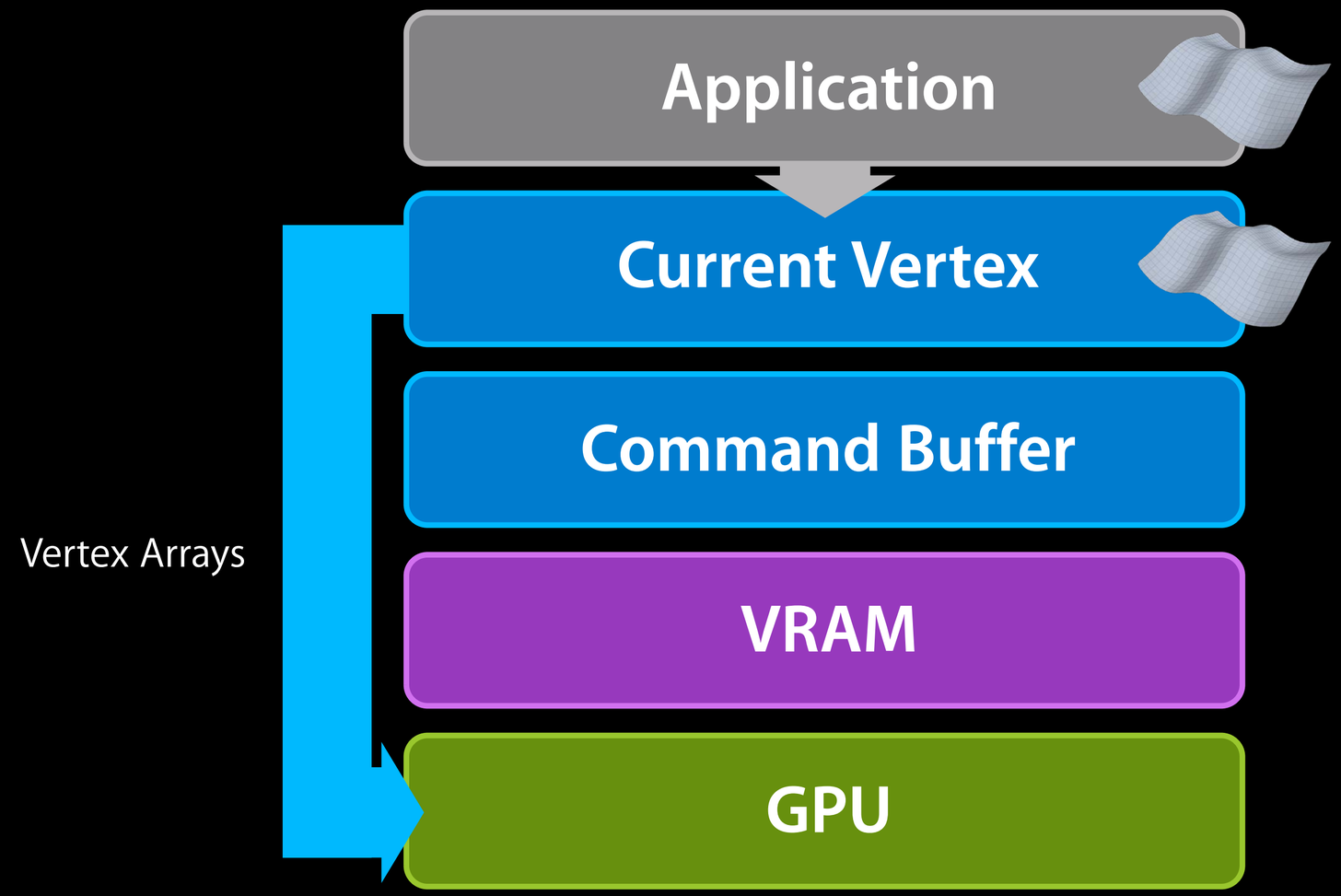
Sample Code: Vertex Performance Test

Vertex Performance Demo

John Stauffer
OpenGL Engineering Manager

http://developer.apple.com/wwdc2005/

Vertex Arrays



Sample Code: Vertex Performance Test

Dynamic Vertex Data

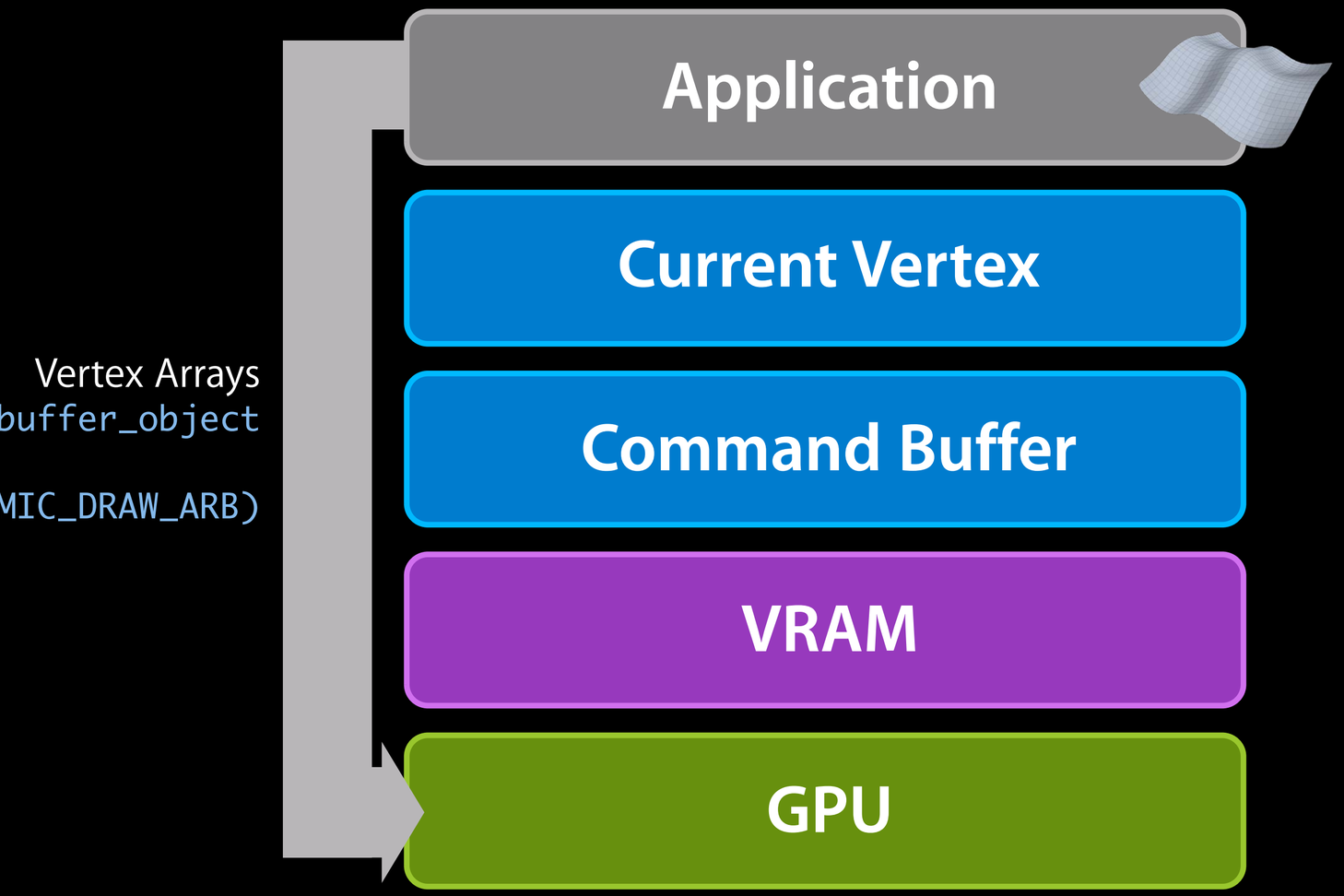
- GL_ARB_vertex_ buffer_object

  GL_DYNAMIC_DRAW_ARB

Sample Code: Vertex Performance Test

Vertex Array Range

Dynamic Data Setup



Vertex Arrays
`GL_ARB_vertex_buffer_object`

`(GL_DYNAMIC_DRAW_ARB)`

Sample Code: Vertex Performance Test

Vertex Buffer Object

Dynamic data setup

```
#define BUFFER_OFFSET(i) ((char *)NULL + (i))

glBindBufferARB(GL_ARRAY_BUFFER_ARB, name);

glEnableClientState(GL_VERTEX_ARRAY);

glVertexPointer(3, GL_FLOAT, stride,BUFFER_OFFSET(0));

glBufferDataARB(GL_ARRAY_BUFFER_ARB, dataSize, NULL
   GL_DYNAMIC_DRAW_ARB);

dest = glMapBufferARB(GL_ARRAY_BUFFER_ARB, GL_WRITE_ONLY_ARB);

<write vertex data into dest>

glUnmapBufferARB(GL_ARRAY_BUFFER_ARB);
```

Sample Code: Vertex Performance Test

Static Vertex Data

- `GL_ARB_vertex_ buffer_object`
  `GL_STATIC_DRAW_ARB`
- Display lists
  - `glBegin/glEnd`
  - Use consistent vertex data
  - Must have at least 16 vertices per draw command

Sample Code: Vertex Performance Test

Vertex Performance Demo

John Stauffer
OpenGL Engineering Manager

http://developer.apple.com/wwdc2005/

More Information

http://developer.apple.com/wwdc2005/

Documentation, Sample Code, and Other Resources

http://developer.apple.com/wwdc2005/

OpenGL Org Web Page

http://www.opengl.org

OpenGL Org Web Page

- http://www.opengl.org

OpenGL Sessions and Labs

Maximizing Mac OS X OpenGL Lab	Tiger Lab Thursday 3:30PM
OpenGL and Quartz Composer Drop-in Lab	Graphics & Media Lab Friday 9:00AM

Who to Contact

John Stauffer
OpenGL Engineering Manager
stauffer@apple.com

Travis Brown
Graphics and Imaging Evangelist
travis@apple.com

Q&A

Travis Brown
Graphics and Imaging Evangelist
travis@apple.com

http://developer.apple.com/wwdc2005/

WWDC2005

Worldwide Developers Conference